An Introduction to Public-Key Cryptography and Infrastructure

William E. Johnston[1]

Information and Computing Sciences Division

Ernest Orlando Lawrence Berkeley National Laboratory

University of California

---

1. **wejohnston@lbl.gov, 510-486-5014, mudumbai@george.lbl.gov, mrt@george.lbl.gov - http://www-itg.lbl.gov**

Encryption

- Encryption is a *key-based* mathematical transformation that changes *plaintext* to *ciphertext* in such a way that the reverse operation - decryption - is very difficult without possession of the key

- Decryption is the inverse transformation, and reverses the encryption (converts ciphertext back to plaintext)
  - reversing the encryption requires you to have, or can guess, the decryption key
  - typically the larger the key, the harder it is to guess

♦ **Two basic kinds of cryptographic transformations**

- **single key**
  - **“secret key” / “symmetric” cryptography**
  - **same key encrypts and decrypts**
  - **e.g. NIST Data Encryption Standard (DES)**
- **two key**
  - **“public key” / “asymmetric” cryptography**
  - **what one key encrypts and a different one decrypts**
  - **e.g. RSA, Diffie-Hellman (NIST Digital Signature Standard (DSS))**

Public-Key Systems

- Keys are generated in matching pairs - typically called a "public key" and a "private key"
  - what one key encrypts, *only* the other can decrypt, and visa versa
- The public key is made "freely" available
  - the private key "cannot" be determined from the public key
- The private key is kept secret
  - only one entity - the "user" or "owner" knows the private key

Uses of Public Key Cryptography

- Private messages without prior secret key exchange
- Document integrity
- Authenticated messages (non-repudiation)
- Confidential data transfer (“bulk” encryption)
- Assured identity and user authentication without “passwords in the clear”
- Digital Timestamp Service - issues timestamps which associate a date and time with a digital document in a cryptographically strong manner

Uses of Public Key Cryptography

Private Messages ("many to one" private communication)

- **Example:**

Entity "A" - a researcher whose workstation is on an open, insecure, network - wishes to communicate with assured privacy with entity "B", a collaborator on the other side of the country

- **B's public key is available from a source that is "reliable"**

  E.g. an employer's X.500/LDAP or Web server (commercial enterprise public keys are frequently published in the business section of the Sunday New York Times)

- **A obtains B's public key over the open network**

- **A uses B's public key to encrypt a message for B**

Uses of Public Key Cryptography

- **The only way to decrypt this message is to use B's private key (which B protects carefully)**
- **Therefore A just uses open e-mail to send the encrypted message to B, who can then decrypt and read it**

How can A be sure that it was really B's key that was obtained? (Through the use of trusted third parties to guarantee the public key ownership.)

Uses of Public Key Cryptography

Document Integrity

- Is what B received exactly what A sent?
  - A generates a message / document
  - A generates a “message digest” (a small, fixed-length code (bit string) that is unique to the original message - a “one-way hash” code)
  - A encrypts the message digest with A’s private key and includes the encrypted digest with the plaintext message / document
  - Anyone can read the original message (e.g., a deed, a computer security advisory message, etc.)
  - B verifies the *integrity* of A’s message by generating the hash code for the received message, decrypting the message digest with A’s public key and comparing the two

(a match => received message is the same as the original)

- ♦ **Other types of document integrity**
  - a "document" can be any digital data (e.g. a digital image, digital video clip, digital audio clip, etc.)
  - a verified time stamp can prove *when* a document was generated
    - "postmarks" - time stamps that verify the contents of a document at a given point in time - can be generated by sending the message digest to a trusted third party who adds a guaranteed accurate time stamp and digitally signs the combined document (digest plus time stamp) and returns it

Uses of Public Key Cryptography

- the US Postal Service will start offering this service commercially in early 1997
- important for patent notes, lab notebooks, etc.

Uses of Public Key Cryptography

Authenticated Messages (“who really sent this message?”)

- **Ciphertext that can be decrypted (converted to plaintext) with a given public key *could only have been encrypted with the corresponding private key***
- **Therefore, if you get a message and can decrypt it with B’s public key, then only the holder of B’s private key could have encrypted it**
- **This allows you to verify the identify of the sender of a message**

Uses of Public Key Cryptography

Confidential Data ("how to exchange lots of private data")

- ♦ **Public key encryption is relatively expensive, and is not suited to encrypting large volumes of data**

- ♦ **Single key encryption (e.g. DES) is suitable for "bulk" encryption**

- ♦ **Example: transferring a confidential file**
  - **Using symmetric / secret key cryptography, a temporary ("session") key is generated and used to encrypt large datasets**
  - **The session key can be encrypted with the public key of the intended recipient and sent, in confidence, to the recipient**
  - **The encrypted data can be transferred over an insecure network**

Uses of Public Key Cryptography

- **The intended recipient decrypts the session key using his/her private key, and then uses the session key to decrypt the data**

Uses of Public Key Cryptography

User Authentication (logging in remotely without passwords)

- If a remote system can obtain a public key for which it has some confidence as to the "identity" of the "person" to whom that public key was issued ....

  ... then there are protocols that allow the remote system to authenticate the user on his/her local system and permit a "login" (e.g. a telnet or X-window session from the local to the remote system) without exchanging any passwords "in the clear" (and over an encrypted channel, if desired)

- See, e.g., "SSH (Secure Shell) Remote Login Program" (http://www.cs.hut.fi/ssh)

“Identity”

- ♦ **Several of the uses of public key cryptography (e.g. user authentication) depend on establishing the identify of the holder of a private key**

- ♦ **That is, how do you know that the holder of the private key (the person that encrypted the message that you have just received and were able to decrypt with the corresponding public key)** ***is really the person that you think it is?***

  - **all that public key cryptography “guarantees” is that public and private keys come in uniquely associated pairs, not who holds the private key**

Establishing Identity

- The "real" identity of the person who controls the private key is established by having a "trusted third party" verify the identify of the key holder - either when the key pairs are issues, or after the fact
- A "certification authority" is like a state DMV issuing a driving license (when you use it as proof of identity)
  - you go to the trusted third party, and present some proof of you identity (e.g. a birth certificate)
  - the third party verifies this proof, and issues a signed certificate that identifies "you" (i.e. with your picture on the license)
  - this certificate is then used as a guarantee that you the person, your name, and your written signature, are all "equivalent"

Establishing Identity

- ♦ **Public key certification authorities (CA) work the same way**
  - **you present a “proof” of identity**
  - **the certification authority generates a public-private key pair**
  - **the private key is issued to “you”**
  - **your public key is placed in a digital document (a “certificate”) that is then signed by the CA, and “published” in a public place (e.g. an X.500 name server or a Web server)**

Establishing Identity

- now when someone uses your public key from this “public key certificate” they have the guarantee of the CA that it represents “you”
  - the strength of the identity guarantee is a function of the procedures used by the CA to establish your “real” identity
  - these procedures are usually published as part of the policy of the CA

Establishing Identity

♦ **CA Issues**

- **why do you trust the trusted third party (CA)?**
- **there may be many CAs with certificates that you need**
- **CAs should probably be "local" (so people who are registering can present themselves in person)**
- **how are different policies represented?**
  - **by different CAs?**
  - **by different or multiple attributes (auxiliary information kept in some types of public key certificates)?**

Establishing Identity

- There are other “trust” models: E.g. the PGP “web of trust” idea
  - you generate your own public-private key pair
  - you have “respected” acquittances, know officials, etc., sign your public key
  - you then publish your public key and its “endorsements”
  - then parties who also know and respect these co-signers of your public key, will accept it as validly representing you

Certification Authorities as Trusted Third Parties

- ♦ The original idea for validating CAs was that the CAs themselves were verified by having their public key signed by a "higher" authority, etc.
  - this is the PEM model described in "Privacy Enhancement for Internet Electronic Mail: Part II: Certificate-Based Key Management" (RFC-1422)
  - the problem with this model is that the signers of CA public keys are "policy certification authorities" (PCAs), and some level of policy uniformity has to be accepted by all of the CAs who's keys are signed by the PCA
    - if the CAs represent large organizations, such common policy is hard to agree on

Establishing Identity

- Current trend for verifying CAs seems to be for CAs to cross sign each other's certificates, on a "pair-wise" basis, and for limited common policy
  - the common policy might include such operational matters as
    - the procedures for running the CA itself (if the CA's private key is compromised, then all subordinate keys will have be reissued)
    - the mechanism for establishing the identity of persons to whom public-private key pairs are issued (e.g. does the applicant have to show up in person, with a valid driver's license, in order to have the CA sign a public key certificate)
    - how long are certificates valid
    - how are certificates revoked

Other Issues

- Key archiving and escrowing
  - if you encrypt corporate data and store it that way, the corporation may feel that it should have a copy of the key
- Private key security
  - people and applications
  - software and hardware key management
- Legal concerns
  - See "Federal Certification Authority Liability and Policy: Law and Policy of Certificate-Based Public Key and Digital Signatures", Michael S. Baum, Published by: U. S. Dept. of Commerce, NIST, June, 1994

Why PKI?

- PKI is seen as the most effective and cost effective (easily administered) technology for providing secure and authenticated transfer of digital information
- PKI scales with a large number of users, organizations, sites, and applications
- PKI is being adopted as the security basis of most network-based commercial services
  - E.g. electronic commerce - MasterCard, Visa, Netscape and Microsoft, and on-line banking - Wells Fargo, BankAmerica, etc.

Other Issues

For more information see:

- **"A Cryptography Primer" (cryptography for commerce - http://www.courttv.com/seminars/handbook/crypto.html)**
- **"Frequently Asked Questions: Cryptography -- The Latest from RSA Labs" (a technical overview - http://www.rsa.com/PUBS)**
- **"Computer Communications Security" Warwick Ford (Prentice-Hall, 1994 - comprehensive textbook)**
-